7U

UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/712,619	11/12/2003	Andrew L. Bliss	MSFT121515	9269

26389 7590 03/21/2007 CHRISTENSEN, O'CONNOR, JOHNSON, KINDNESS, PLLC 1420 FIFTH AVENUE SUITE 2800 SEATTLE, WA 98101-2347	**EXAMINER** MCCARTHY, CHRISTOPHER S

ART UNIT	PAPER NUMBER
2113	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	03/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

PTOL-90A (Rev. 10/06)

	Application No.	Applicant(s)
Office Action Summary	10/712,619	BLISS ET AL.
	Examiner	Art Unit	
	Christopher S. McCarthy	2113	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on *22 January 2007*.

2a) ☒ This action is **FINAL**.   2b) ☐ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) *1-32* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) *1,2,4-6,8-15,17-19 and 21-32* is/are rejected.

7) ☒ Claim(s) *3,7,16 and 20* is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☒ The drawing(s) filed on *11/n/05* is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a) ☐ All   b) ☐ Some *   c) ☐ None of:

      1. ☐ Certified copies of the priority documents have been received.

      2. ☐ Certified copies of the priority documents have been received in Application No. _____.

      3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
    Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
    Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☒ Other: *response to arguments*.

DETAILED ACTION

1.   Claims 25-32 are rejected under 35 U.S.C. 101 because the claimed invention is directed

to non-statutory subject matter, as cited in prior office action, which was mailed on 7/21/06.

2.   Claims 1-2, 4-6, 8-15, 17-19, 21-30, 32 are rejected under 35 U.S.C. 102(e) as being

anticipated by Hendel et al. U.S. Patent 7,028,056, as cited in prior office action, which was

mailed on 7/21/06.

3.   Claims 3, 7, 16, 20 are objected to as being dependent upon a rejected base claim, but

would be allowable if rewritten in independent form including all of the limitations of the base

claim and any intervening claims, as cited in prior office action, which was mailed on 7/21/06.

Claim Rejections - 35 USC § 101

4.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or
> any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and
> requirements of this title.

Claims 25-32 are rejected under 35 U.S.C. 101 because the claimed invention is directed

to non-statutory subject matter.   While the applicant has amended the claim language to include

a processor and a memory, there is no claim language in which the operating environment is

stored in the memory and executed by the processor. The coexistence of software among

hardware components is not enough to overcome the non-statutory rejection.

Claim Rejections - 35 USC § 102

5.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form

the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

6.   Claims 1-2, 4-6, 8-15, 17-19, 21-30, 32 are rejected under 35 U.S.C. 102(e) as being

anticipated by Hendel et al. U.S. Patent 7,028,056.

As per claim 1, Hendel teaches in a computer system having an operating environment

including user mode modules having a first level of protection and kernel mode modules having

a second level of protection (column 2, lines 32-35; column 5, lines 5, lines 29-33, wherein

Hendel teaches that user-mode portion processes have higher accessibility/lower protection than

kernel mode (kernel mode not available to the user)), a method for consistently collecting

information associated with the execution of a user mode module (column 1, lines 6-10), the

method comprising: transmitting, by a requester application, a request to collect kernel mode

module information (column 3, lines 27-49, wherein Hendel teaches that method includes a

client process/application that issues/requests a write and read dump), wherein the request to

collect kernel mode module information includes an identification of one or more executing

process threads from which kernel mode information will be collected (column 3, lines 27-49;

column 2, lines 41-53, wherein Hendel teaches a dump to include parameters and teaches the parameters/information to include thread context information); obtaining, by a kernel mode module, the request to collect kernel mode module information (column 2, liens 41-53; column 6, lines 20-29); capturing, by the kernel mode module, information corresponding to each thread identified in the request to collect kernel mode module information (column 2, lines 41-53, 58-61; column 6, lines 20-28); transmitting, by the kernel mode module, a result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information (column 7, lines 9-21; column 3, lines 21-26); and receiving, by the requestor application, the result of the capturing of the information corresponding to each thread identified in the request to collect kernel mode module information (column 3, lines 37-48; column 7, lines 15-21).

As per claim 2, Hendel teaches the method as recited in claim 1, wherein the request to capture kernel mode module information includes an identification of a pre-allocated memory in which to store captured kernel mode information (column 3, lines 46-48; column 6, lines 42-49; column 7, lines 15-21).

As per claim 4, Hendel teaches the method as recited in claim 1, wherein the kernel mode module is an operating system resident application (figure 6; column 5, lines 22-43).

As per claim 5, Hendel teaches the method as recited in claim 1 further comprising capturing, by the kernel mode module, a list of all loaded drivers (column 6, lines 23-26).

As per claim 6, Hendel teaches the method as recited in claim 1, wherein capturing information corresponding to each thread identified in the request to collect kernel mode module information includes: (a) capturing a thread kernel stack (column 6, line 16); and (b) capturing

all pending I/O request packet information (column 6, lines 33-37, 15; column 5, lines 24-25,

wherein, combining these citations, the dump contains information about the crashing process

that was running on the crashing driver at the time of failure; therefore, since the kernel is

running a device driver, that inherently is used to process data from one device to another, and

that driver crashes, then the information about the process that was running on the driver, that is,

the I/O request, is saved in the dump as part of the information of the crashing process on the

crashing driver); and (c) repeating (a)-(b) for each identified thread in the request to capture

kernel mode module information (column 2, lines 49-52, wherein, multiple threads/process

information is saved; therefore, the taught thread information taught as saved for each thread, by

Hendel, is saved the same for multiple threads).

As per claim 8, Hendel teaches the method as recited in claim 6, wherein capturing

information corresponding to each thread identified in the request to collect kernel mode module

information is asynchronous (column 8, lines 65-67, wherein a request could happen at anytime;

there is no teaching of being dependent upon time).

As per claim 9, Hendel teaches the method as recited in claim 1, wherein transmitting a

result includes transmitting a status code corresponding to the success or failure of the

information capture (column 3, lines 43-44, 35-36).

As per claim 10, Hendel teaches the method as recited in claim 1, wherein transmitting a

result includes storing the captured kernel mode module information in an allocated memory

(column 7, lines 9-21).

As per claim 11, Hendel teaches the method as recited in claim 1, wherein transmitting a request to collect kernel mode module information occurs in response to a user mode module error (column 7, lines 37-40).

As per claim 12, Hendel teaches a computer-readable medium having computer-executable instructions for performing the method recited in claim 1 (column 4, lines 28-48; see rejection of claim 1).

As per claim 13, Hendel teaches a computer system having a processor, a memory and an operating environment, the computer system for performing the method recited in claim 1 (column 4, lines 15-48; see rejection of claim 1).

As per claim 14, Hendel teaches in a computer system having an operating environment including user mode modules having a first level of protection and kernel mode modules having a second level of protection (column 2, lines 32-35; column 5, lines 5, lines 29-33, wherein Hendel teaches that user-mode portion processes have higher accessibility/lower protection than kernel mode (kernel mode not available to the user)), a method for consistently collecting information associated with the execution of a user mode module, the method comprising: obtaining a user mode module request to collect kernel mode module information including an identification of one or more executing process threads from which kernel mode information will be collected (column 2, lines 41-53; column 6, lines 20-29; column 3, lines 27-49; column 2, lines 41-53, wherein Hendel teaches a dump to include parameters and teaches the parameters/information to include thread context information); capturing information corresponding to each thread identified in the request to collect kernel mode module information

; and transmitting the captured kernel mode module information (column 2, lines 41-53, 58-61; column 6, lines 20-28; column 7, lines 9-21; column 3, lines 21-26).

As per claim 15, Hendel teaches the method as recited in claim 14, wherein the request to capture kernel mode module information includes an identification of a pre-allocated memory in which to store captured kernel mode information (column 3, lines 46-48; column 6, lines 42-49; column 7, lines 15-21).

As per claim 17, Hendel teaches the method as recited in claim 14, wherein obtaining a user mode module request includes obtaining, by an operating system resident application, the user mode module request (column 5, lines 22-43; figure 6).

As per claim 18, Hendel teaches the method as recited in claim 14 further comprising capturing a list of all loaded drivers (column 6, lines 23-26).

As per claim 19, Hendel teaches the method as recited in claim 14, wherein capturing information corresponding to each thread identified in the request to collect kernel mode module information includes: (a) capturing a thread kernel stack (column 6, line 16); and (b) capturing all pending I/O request packet information (column 6, lines 33-37, 15; column 5, lines 24-25, wherein, combining these citations, the dump contains information about the crashing process that was running on the crashing driver at the time of failure; therefore, since the kernel is running a device driver, that inherently is used to process data from one device to another, and that driver crashes, then the information about the process that was running on the driver, that is, the I/O request, is saved in the dump as part of the information of the crashing process on the crashing driver); and (c) repeating (a)-(b) for each identified thread in the request to capture kernel mode module information (column 2, lines 49-52, wherein, multiple threads/process

information is saved; therefore, the taught thread information taught as saved for each thread, by

Hendel, is saved the same for multiple threads).

As per claim 21, Hendel teaches the method as recited in claim 19, wherein capturing

information corresponding to each thread identified in the request to collect kernel mode module

information is asynchronous (column 8, lines 65-67, wherein a request could happen at anytime;

there is no teaching of being dependent upon time).

As per claim 22, Hendel teaches the method as recited in claim 1, wherein transmitting

the captured kernel mode module information includes transmitting a status code corresponding

to the success or failure of the information capture (column 3, lines 43-44, 35-36).

As per claim 23, Hendel teaches a computer-readable medium having computer-

executable instructions for performing the method recited in claim 14 (column 4, lines 15-45; see

rejection of claim 14).

As per claim 24, Hendel teaches a computer system having a processor, a memory and an

operating environment, the computer system for performing the method recited in claim 14

(column 4, lines 15-45; see rejection of claim 14).

As per claim 25, Hendel teaches in a computer system having a processor, a memory, and

an operating environment, the operating environment including user mode modules having a first

level of protection and kernel mode applications having a second level of protection (column 2,

lines 32-35; column 5, lines 5, lines 29-33, wherein Hendel teaches that user-mode portion

processes have higher accessibility/lower protection than kernel mode (kernel mode not available

to the user)), a software architecture for consistently collecting information associated with the

execution of a user mode module, the architecture comprising: a processing component for

capturing kernel mode module information corresponding to one or more executing processing threads identified in a request to collect kernel mode module information (column 2, lines 41-53, 58-61; column 6, lines 20-28; ; and at least one application program interface for accessing the processing component and identifying the one or more processing threads from which to collect kernel mode module information (column 3, lines 21-26).

As per claim 26, Hendel teaches the architecture as recited in claim 25, wherein the at least one application program interface is further operable to identify a pre-allocated memory to received captured kernel mode module information (column 3, lines 46-48; column 6, lines 42-49; column 7, lines 15-21).

As per claim 27, Hendel teaches the architecture as recited in claim 25, wherein the processing component is embodied as a driver application (column 6, lines 64-67).

As per claim 28, Hendel teaches the architecture as recited in claim 25, wherein the processing component is embodied as an operating system resident application (column 5, lines 22-43; figure 6).

As per claim 29, Hendel teaches the architecture as recited in claim 25, wherein the kernel mode module information includes a list of all loaded drivers (column 6, lines 23-26).

As per claim 30, Hendel teaches the architecture as recited in claim 25, wherein the kernel mode module information includes a thread kernel stack (column 6, line 16) and all pending I/O request packet information for each identified process thread (column 6, lines 33-37, 15; column 5, lines 24-25, wherein, combining these citations, the dump contains information about the crashing process that was running on the crashing driver at the time of failure; therefore, since the kernel is running a device driver, that inherently is used to process data from

one device to another, and that driver crashes, then the information about the process that was

running on the driver, that is, the I/O request, is saved in the dump as part of the information of

the crashing process on the crashing driver).

As per claim 32, Hendel teaches the architecture as recited in claim 25, wherein the

process component captures the kernel mode module information asynchronously (column 8,

lines 65-67, wherein a request could happen at anytime; there is no teaching of being dependent

upon time).

The applied reference has a common assignee with the instant application. Based upon

the earlier effective U.S. filing date of the reference, it constitutes prior art under 35 U.S.C.

102(e). This rejection under 35 U.S.C. 102(e) might be overcome either by a showing under 37

CFR 1.132 that any invention disclosed but not claimed in the reference was derived from the

inventor of this application and is thus not the invention "by another," or by an appropriate

showing under 37 CFR 1.131.

Allowable Subject Matter

7.   Claims 3, 7, 16, 20 are objected to as being dependent upon a rejected base claim, but

would be allowable if rewritten in independent form including all of the limitations of the base

claim and any intervening claims.

Response to Arguments

8.   Applicant's arguments filed 1/22/07 have been fully considered but they are not persuasive.

With respect to claims 1, 14, and 25, the applicant has amended and argues the claim language of the kernel mode module information including an identification of one or more executing process threads. The applicant has argued that the newly added "executing" language overcomes the art of Hendel. The examiner respectfully disagrees. Simply, Hendel teaches, in column 2, lines 58-62 and column 3, line 6, the identification of the thread executing when a crash occurs. The examiner deems this sufficient to reject the newly added "executing" language, in that, the process was the executing thread at the time of collection. If the applicant would like the claims to read as the thread continues to execute during collection, then the claim language should reflect that, providing it is enabled by the specification.

With respect to claim 8, the applicant has argued that Hendel does not teach the capturing of information to be asynchronous, that is, at any time. The examiner respectfully disagrees. As stated above, the collection can be upon a failure or a crash of a process thread; however, Hendel also teaches, in column 8, lines 63-65, that the collection can occur at any time other than when a crash has occurred; that is, it can collect the information at time of a crash or at a time other than a crash; this is interpreted by the examiner as asynchronous or anytime.

In light of the above arguments, all applicable rejected claims stand.

Conclusion

9.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time

policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE

MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period

will expire on the date the advisory action is mailed, and any extension fee pursuant to 37

CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing

date of this final action.

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Christopher S. McCarthy whose telephone number is (571)272-

3651. The examiner can normally be reached on M-F, 9 - 5:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Robert Beausoliel can be reached on (571)272-3645. The fax phone number for the

organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would

like assistance from a USPTO Customer Service Representative or access to the automated

information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

csm